

BLM NAVIGATOR

Geocommunicator Web Services Migration Guide

Dec 5, 2016

GeoCommunicator Web Services Migration Guide

The GeoCommunicator Web Services were written using a SOAP API, a programming interface using standardized XML documents to register programmatic objects and data that can be passed between computer programs. SOAP, as a web technology, has been generally surpassed by REST, which uses HTTP verbs and URL endpoints to communicate data, usually in a JavaScript object (JSON) format. While the general operations and parameter names of the GeoCommunicator Web Services will remain the same, access to them and expected input/output data formats have been drastically changed.

For detailed information about the input and output data formats for the new Navigator Special Services operations, including detailed examples, please check the BLM Navigator Special Services User Guide located at <https://navigator.blm.gov/api/resources/guides/BLM-SpecialServicesUserGuide.pdf>.

Find Land Description

Endpoints:

GeoCommunicator published five FindLD operations: FindLD, FindLD2, LDToGeoRSS, LDToGeoRSS2, LDToGeoRSS3. The LDToGeoRSS operations have been deprecated. The only difference between FindLD and FindLD2 was that FindLD2 contained an InputFormat parameter specifying the format entered in the LegalDescription parameter; as FindLD already interprets several different input formats and this parameter is the only difference between the two operations, the FindLD2 operation has been deprecated.

Navigator publishes one FindLD operation. This operation functions effectively in the same manner as FindLD.

Inputs:

legaldescription (string) – a series of characters that describe a parcel of land according to the Public Land Survey System (PLSS). This maps directly to the LegalDescription input parameter for GeoCommunicator FindLD.

Outputs:

legaldescription (string) – this is the exact text the user entered in the legaldescription input field. This maps directly to the LegalDescription XML tag in the output for GeoCommunicator FindLD.

generatedplss (array) – this is a series of strings listing the PLSS IDs generated by parsing the legaldescription input field. This does not map to any of the GeoCommunicator FindLD output data items.

status (bool) – this field determines if the operation completed with no errors. This maps directly to the CompletionStatus XML tag in the output for GeoCommunicator FindLD.

features (array) – this is a series of JSON objects conforming to the official ESRI format for spatial feature

data, containing a geometry object (which conforms to the ESRI format for spatial geometry) and an attributes object. The attributes object contains a legaldescription property containing the PLSS ID for that specific feature. This maps directly to the Polygons XML tag in the output for GeoCommunicator FindLD.

Example (HTTP POST):

```
$.ajax({
  url: "https://path/to/service/FindLD",
  method: "POST",
  data: { legaldescription: "NV 21 T38N R56E SEC 10 ALIQ SESW" }
})
.done(function(data) {
  // data contains properties listed above in Outputs section
})
```

Example (HTTP GET):

<https://path/to/service/FindLD?legaldescription=N%2021%20T38N%20R56E%20SEC%2010%20ALIQ%20SESW>

Derive Land Description

Endpoints:

GeoCommunicator published five DeriveLD operations: DeriveLD, DeriveLD2, DeriveLDToGeoRSS, DeriveLDToGeoRSS2, DeriveLDToGeoRSS3. The DeriveLDToGeoRSS operations have been deprecated. The only differences between DeriveLD and DeriveLD2 was that DeriveLD2 contained an eLDFormat parameter specifying the format the LegalDescription output tag was returned as, and a SpatialRel parameter specifying whether to return items that intersected or were completely contained by the geometry specified in the SelectionGeometry input parameter. As these parameters have very minor effect on the output data, the DeriveLD2 operation was deprecated.

Navigator publishes one DeriveLD operation. This operation functions effectively in the same manner as DeriveLD.

Inputs:

selectiongeometry (object) – a JSON object containing a features array that contains one or more JSON objects that conform to the official ESRI format for spatial feature data. The attributes values of feature objects are ignored for the purposes of the DeriveLD operation – only the geometry is parsed. This maps directly to the SelectionGeometry input parameter for GeoCommunicator DeriveLD.

Outputs:

selectiongeometry (object) – this is the exact JSON the user entered in the selectiongeometry input field. This maps directly to the Polygons XML tag in the output for GeoCommunicator DeriveLD.

status (bool) – this field determines if the operation completed with no errors. This maps directly to the

CompletionStatus XML tag in the output for GeoCommunicator DeriveLD.

landdescriptions (array) – this is a series of strings listing the PLSS IDs found that intersect with the geometry of the features submitted into the selectiongeometry input parameter. This maps directly to the LegalDescription XML tag in the output for GeoCommunicator DeriveLD.

Example (HTTP POST):

```
$.ajax({
  url: "https://path/to/service/DeriveLD",
  method: "POST",
  data: { selectiongeometry: {
    "features": [
      {
        "geometry": {
          "rings": [
            [
              [
                -12870483.526700001,
                5042570.563000001
              ],
              [
                -12870480.2687,
                5043106.2224000022
              ],
              [
                -12869948.0976,
                5043112.3192000017
              ],
              [
                -12869951.175999999,
                5042578.0938000008
              ],
              [
                -12870483.526700001,
                5042570.563000001
              ]
            ]
          ],
          "spatialReference": {
            "wkid": 102100,
            "latestWkid": 3857
          }
        }
      }
    ]
  }}
})
.done(function(data) {
  // data contains properties listed above in Outputs section
})
```

Example (HTTP GET):

```
https://path/to/service/DeriveLD?selectiongeometry={"features":[{"geometry":{"rings":[[[-12870483.526700001,5042570.563000001],[-12870480.2687,5043106.2224000022],[-12869948.0976,5043112.3192000017],[-12869951.175999999,5042578.0938000008],[-12870483.526700001,5042570.563000001]]],"spatialReference":{"wkid":102100,"latestWkid":3857}}}]}}
```

Township Geocoder Description

Endpoints:

GeoCommunicator published seven TownshipGeocoder operations: GetLatLon, GetLatLonFeed, GetPMList, GetStateList, GetTRS, GetTRSFeed, ProcessBatchData. GetLatLonFeed is the same operation as GetLatLon, but its output is an XML GeoRSS feed rather than a SOAP encapsulated GeoRSS feed. GetTRSFeed is the same operation as GetTRS, but its output is an XML GeoRSS feed rather than a SOAP encapsulated RSS feed. GetPMList, GetStateList, and ProcessBatchData are all utility operations that are not meant for public use. The GetLatLonFeed, GetTRSFeed, GetPMList, GetStateList, and ProcessBatchData operations have been deprecated.

Navigator publishes GetLatLon and GetTRS operations. These operations function effectively in the same manner as the GetLatLon and GetTRS GeoCommunicator operations.

GetLatLon Inputs:

trs (string) – a series of characters that describe a parcel of land according to the Public Land Survey System (PLSS). This maps directly to the TRS input parameter for GeoCommunicator GetLatLon.

GetLatLon Outputs:

trs (string) – this is the exact text the user entered in the trs input field. This maps indirectly to the encapsulated GeoRSS feed data output by GeoCommunicator GetLatLon.

generatedplss (array) – this is a series of strings listing the PLSS IDs generated by parsing the legaldescription input field. This does not map to any of the GeoCommunicator GetLatLon output data items.

status (bool) – this field determines if the operation completed with no errors. This maps directly to the CompletionStatus XML tag in the output for GeoCommunicator GetLatLon.

features (array) – this is a series of JSON objects conforming to the official ESRI format for spatial feature data, containing a geometry object (which conforms to the ESRI format for spatial geometry) and an attributes object. The attributes object contains a legaldescription property containing the PLSS ID for that specific feature. This maps indirectly to the encapsulated GeoRSS feed data output by GeoCommunicator GetLatLon.

Example (HTTP POST):

```
$.ajax({
  url: "https://path/to/service/GetLatLon",
  method: "POST",
  data: { trs: "NV 21 T38N R56E SEC 10 ALIQ SESW" }
})
.done(function(data) {
  // data contains properties listed above in Outputs section
})
```

Example (HTTP GET):

<https://path/to/service/GetLatLon?trs=NW> 21 T38N R56E SEC 10 ALIQ SESW

GetTRS Inputs:

lat (string) – the latitude or Y coordinate of a geospatial point in Mercator projection. This value may be in decimal degrees or space-separated degrees/minutes/seconds format. This maps directly to the Lat input parameter for GeoCommunicator GetTRS.

lon (string) – the longitude or X coordinate of a geospatial point in Mercator projection. This value may be in decimal degrees or space-separated degrees/minutes/seconds format. This maps directly to the Lon input parameter for GeoCommunicator GetTRS.

units (string) – this field instructs the GetTRS operation how to interpret the lat/lon input coordinates: “DD” for decimal degrees or “DMS” for degrees/minutes/seconds. This maps directly to the Units parameter for GeoCommunicator GetTRS.

GetTRS Outputs:

lat (string) – this is the exact text the user entered in the lat input field. This maps indirectly to the encapsulated GeoRSS feed data output by GeoCommunicator GetTRS.

lon (string) – this is the exact text the user entered in the lon input field. This maps indirectly to the encapsulated GeoRSS feed data output by GeoCommunicator GetTRS.

units (string) – this is the exact text the user entered in the units input field. This does not map to any of the GeoCommunicator GetTRS output data items.

status (bool) – this field determines if the operation completed with no errors. This maps directly to the CompletionStatus XML tag in the output for GeoCommunicator GetTRS.

features (array) – this is a series of JSON objects conforming to the official ESRI format for spatial feature data, containing a geometry object (which conforms to the ESRI format for spatial geometry) and an attributes object. The attributes object contains a legaldescription property containing the PLSS ID for that specific feature. This maps indirectly to the encapsulated GeoRSS feed data output by GeoCommunicator GetTRS.

Example (HTTP POST):

```
$.ajax({
  url: "https://path/to/service/GetTRS",
  method: "POST",
  data: { lat: "41.206456323113024", lon: "-115.61511640360382", units: "DD" }
})
.done(function(data) {
  // data contains properties listed above in Outputs section
})
```

Example (HTTP GET):

<https://path/to/service/GetTRS?lat=41.206456323113024&lon=-115.61511640360382&units=DD>